



# **NAVAL POSTGRADUATE SCHOOL**

**MONTEREY, CALIFORNIA**

## **THESIS**

**UNDERWATER CLOUDS: UTILIZING PRIVATE CLOUD  
ARCHITECTURE ABOARD U.S. SUBMARINES**

by

Christopher David Brooks

June 2012

Thesis Advisor:  
Second Reader:

Douglas J. MacKinnon  
Albert Barreto

**Approved for public release; distribution is unlimited**

THIS PAGE INTENTIONALLY LEFT BLANK

<b>REPORT DOCUMENTATION PAGE</b>			<i>Form Approved OMB No. 0704-0188</i>	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.				
<b>1. AGENCY USE ONLY (Leave blank)</b>		<b>2. REPORT DATE</b> June 2012	<b>3. REPORT TYPE AND DATES COVERED</b> Master's Thesis	
<b>4. TITLE AND SUBTITLE</b> Underwater Clouds: Utilizing Private Cloud Architecture Aboard U.S. Submarines			<b>5. FUNDING NUMBERS</b>	
<b>6. AUTHOR(S)</b> Christopher David Brooks				
<b>7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)</b> Naval Postgraduate School Monterey, CA 93943-5000			<b>8. PERFORMING ORGANIZATION REPORT NUMBER</b>	
<b>9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES)</b> N/A			<b>10. SPONSORING/MONITORING AGENCY REPORT NUMBER</b>	
<b>11. SUPPLEMENTARY NOTES</b> The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB Protocol number ____N/A____.				
<b>12a. DISTRIBUTION / AVAILABILITY STATEMENT</b> Approved for public release; distribution is unlimited			<b>12b. DISTRIBUTION CODE</b>	
<b>13. ABSTRACT (maximum 200 words)</b>  <p>The Consolidated Afloat Network and Enterprise Systems (CANES) program was tasked with replacing computer networks afloat, to both improve productivity and functionality of the fleet's Information Technology infrastructure. The program, in its first iteration, fails to include half the ships currently in service and is limited to a small number of specific platforms. Upgrading the existing software on the current platforms may pose a useful resolution to this issue. The computer networks on board submarines provide a good opportunity to explore how cloud computing could benefit older platforms. This thesis presents a proof of concept for the use of a Private Cloud architecture on board U.S. submarines and how improving computer networks may be possible by leveraging the currently installed hardware without requiring a complete system reconfiguration. We use the "Ubuntu Server Private Cloud" as a basic example to illustrate and explore potential benefits and limitations of the Platform as a Service (PAAS) model. The revised system is examined in terms of its application aboard a submarine and explores how it compares to previous network architectures, such as the Client/Server model. The Ubuntu model was chosen for its usability and robust features and because it is open source and free.</p>				
<b>14. SUBJECT TERMS</b> Cloud Computing, Private Cloud, Submarine, CANES, Afloat Network			<b>15. NUMBER OF PAGES</b> 63	
			<b>16. PRICE CODE</b>	
<b>17. SECURITY CLASSIFICATION OF REPORT</b> Unclassified	<b>18. SECURITY CLASSIFICATION OF THIS PAGE</b> Unclassified	<b>19. SECURITY CLASSIFICATION OF ABSTRACT</b> Unclassified	<b>20. LIMITATION OF ABSTRACT</b> UU	

THIS PAGE INTENTIONALLY LEFT BLANK

**Approved for public release; distribution is unlimited**

**UNDERWATER CLOUDS: UTILIZING PRIVATE CLOUD  
ARCHITECTURE ABOARD U.S. SUBMARINES**

Christopher David Brooks  
Lieutenant, United States Navy  
B.A., Florida State University, 2006  
B.S., Florida State University, 2006

Submitted in partial fulfillment of the  
requirements for the degree of

**MASTER OF SCIENCE IN SYSTEMS TECHNOLOGY  
(COMMAND, CONTROL & COMMUNICATIONS)**

from the

**NAVAL POSTGRADUATE SCHOOL  
June 2012**

Author: Christopher David Brooks

Approved by: Dr. Douglas J. MacKinnon  
Thesis Advisor

Albert Barreto  
Second Reader

Dr. Dan Boger  
Chair, Department of Information Sciences

THIS PAGE INTENTIONALLY LEFT BLANK

## **ABSTRACT**

The Consolidated Afloat Network and Enterprise Systems (CANES) program was tasked with replacing computer networks afloat, to both improve productivity and functionality of the fleet's Information Technology infrastructure. The program, in its first iteration, fails to include half the ships currently in service and is limited to a small number of specific platforms. Upgrading the existing software on the current platforms may pose a useful resolution to this issue. The computer networks on board submarines provide a good opportunity to explore how cloud computing could benefit older platforms. This thesis presents a proof of concept for the use of a Private Cloud architecture on board U.S. submarines and how improving computer networks may be possible by leveraging the currently installed hardware without requiring a complete system reconfiguration. We use the "Ubuntu Server Private Cloud" as a basic example to illustrate and explore potential benefits and limitations of the Platform as a Service (PAAS) model. The revised system is examined in terms of its application aboard a submarine and explores how it compares to previous network architectures, such as the Client/Server model. The Ubuntu model was chosen for its usability and robust features and because it is open source and free.

THIS PAGE INTENTIONALLY LEFT BLANK



# TABLE OF CONTENTS

<b>I.</b>	<b>INTRODUCTION .....</b>	<b>1</b>
<b>A.</b>	<b>CLOUD COMPUTING DEFINITIONS .....</b>	<b>1</b>
1.	Deployment Models [12] .....	2
a.	<i>Public Cloud</i> .....	2
b.	<i>Private Cloud</i> .....	3
c.	<i>Community Cloud</i> .....	3
d.	<i>Hybrid Cloud</i> .....	3
2.	Service Models [2] .....	5
a.	<i>Software as a Service (SAAS)</i> .....	5
b.	<i>Platform as a Service (PAAS)</i> .....	5
c.	<i>Infrastructure as a Service (IAAS)</i> .....	5
<b>B.</b>	<b>VIRTUALIZATION.....</b>	<b>6</b>
1.	General Description .....	6
<b>C.</b>	<b>SUBMARINE NETWORKING .....</b>	<b>7</b>
1.	Types of Submarines .....	7
a.	<i>Fast Attack</i> .....	7
b.	<i>Ohio Class</i> .....	8
2.	Basic Network Specifications .....	9
a.	<i>Hardware</i> .....	9
b.	<i>Software</i> .....	9
<b>D.</b>	<b>LITERATURE REVIEW .....</b>	<b>10</b>
1.	Cloud Computing Handbook.....	10
2.	CANES.....	10
3.	NGEN .....	10
4.	Quality of Service Cornerstones .....	10
a.	<i>Efficiency</i> .....	11
b.	<i>Scalability</i> .....	11
c.	<i>Robustness</i> .....	11
d.	<i>Security</i> .....	11
<b>E.</b>	<b>PROBLEM MOTIVATION .....</b>	<b>12</b>
1.	Frustration.....	12
a.	<i>Manpower Costs</i> .....	12
2.	Compatibility .....	13
3.	Standardization .....	13
<b>II.</b>	<b>PROBLEM DESCRIPTION .....</b>	<b>15</b>
<b>A.</b>	<b>HISTORY .....</b>	<b>15</b>
1.	Client Server.....	16
2.	Server Hardware.....	17
3.	Network Hardware .....	17
4.	Clients.....	18
<b>B.</b>	<b>CONTEXT .....</b>	<b>19</b>
1.	Confined Area .....	19

2.	Standardized Equipment and Procedures.....	20
3.	Air-Gap .....	20
4.	Budget Constraints, Older Platforms .....	21
III.	MODEL DESCRIPTION.....	23
A.	ASSUMPTIONS.....	23
1.	Hardware .....	24
a.	<i>Experimental Model</i> .....	24
b.	<i>Submarine</i> .....	25
c.	<i>Squadron/Group</i> .....	26
2.	Software .....	26
3.	Budget .....	27
B.	OUTPUTS AND FINDINGS .....	28
C.	OPERATIONS .....	28
1.	Lab Setup Process .....	28
2.	Accessibility .....	30
D.	DATA .....	30
1.	Experimental model.....	30
a.	<i>Hardware Specifications</i> .....	30
b.	<i>Outputs</i> .....	31
c.	<i>Accessibility</i> .....	31
2.	Potential Sub Model.....	31
a.	<i>Hardware Specifications</i> .....	31
b.	<i>Outputs</i> .....	32
c.	<i>Accessibility</i> .....	32
3.	Potential Squadron/Group Model.....	32
a.	<i>Hardware Specifications</i> .....	32
b.	<i>Outputs</i> .....	33
c.	<i>Accessibility</i> .....	33
IV.	CONCLUSION .....	35
A.	PRIVATE CLOUD PARADIGM.....	35
1.	What Cloud Means .....	35
B.	POTENTIAL SOLUTIONS.....	35
1.	Purchase New Equipment .....	35
2.	Stick with Current Equipment and Operations.....	36
3.	Improvise .....	36
C.	QUALITY OF SERVICE CORNERSTONES .....	36
1.	Efficiency .....	36
2.	Scalability .....	37
3.	Robustness .....	37
4.	Security .....	37
D.	SUMMARY .....	38
V.	RECOMMENDATIONS AND FUTURE RESEARCH .....	39
A.	COST ANALYSIS .....	39
1.	Virtualization.....	39

2.	Technical support Subscriptions .....	39
B.	CLASSIFICATION .....	40
1.	Specific Equipment .....	40
2.	Security Analysis .....	40
LIST OF REFERENCES.....		41
INITIAL DISTRIBUTION LIST .....		43

THIS PAGE INTENTIONALLY LEFT BLANK

## LIST OF FIGURES

Figure 1.	Cloud Computing Deployment Models [From 12].....	2
Figure 2.	Illustration of Enterprise Cloud [From 1] .....	4
Figure 3.	Cloud Computing Service Models [From 2] .....	4
Figure 4.	Illustration of Hypervisor Layer Virtualization [From 4].....	7
Figure 5.	Experimental Cloud Setup .....	25
Figure 6.	Submarine Cloud Model .....	26

THIS PAGE INTENTIONALLY LEFT BLANK

## LIST OF TABLES

Table 1.	Virtual Machine Instance Sizes .....	27
Table 2.	Operational Time Requirements (in Minutes) .....	28
Table 3.	Ubuntu Setup Process with Required Times .....	29
Table 4.	Experimental Model Hardware Specifications .....	30
Table 5.	Experimental Model Output .....	31
Table 6.	Potential Sub Model Specifications .....	31
Table 7.	Potential Submarine Model Output .....	32
Table 8.	Potential Squadron/Group Model .....	32
Table 9.	Potential Squadron/Group Output.....	33

THIS PAGE INTENTIONALLY LEFT BLANK



## **LIST OF ACRONYMS AND ABBREVIATIONS**

C4I	Command, Control, Computers, Communications and Intelligence
CANES	Consolidated Afloat Network Enterprise Systems
COP	Common Operational Picture
COTS	Commercial Off-the-Shelf
CPU	Central Processing Unit
CVN	Nuclear Powered Aircraft Carrier
DDG	Guided Missile Destroyer
EC2	Elastic Compute Cloud
GB	Gigabyte
GPU	Graphics Processing Unit
IAAS	Infrastructure as a Service
IT	Information Technology
NIC	Network Interface Card
NIST	National Institute of Standards and Technology
PAAS	Platform as a Service
PEO	Program Executive Office
RAM	Random Access Memory
SAAS	Software as a Service
SSBN	Nuclear Powered Ballistic Missile Submarine
SSGN	Nuclear Powered Guided Missile Submarine
SSN	Nuclear Powered Attack Submarine
TB	Terabyte
U.S.	United States
VM	Virtual Machine

THIS PAGE INTENTIONALLY LEFT BLANK

## **ACKNOWLEDGMENTS**

I would first like to thank my family for understanding the time required to put together a report like this. My daughter and I enjoyed many long hours at the table on our work; I working on this thesis, and she coloring pictures of princesses and castles.

I would also like to thank my professors, especially Professor Albert “Buddy” Barreto for his enthusiastic inspiration. His class introduced me to many new options in networking and computer systems, and his enthusiasm for talking about new and seemingly mundane computer topics kept me interested.

I would like to thank both of my thesis advisors for the work they have put into helping me finish this project. It has been a long year, and without their help I would not have made it. They are an invaluable source of academic knowledge and support.

THIS PAGE INTENTIONALLY LEFT BLANK

## **I. INTRODUCTION**

“Cloud computing” is a popular term being coined in society today in both technical and non-technical circles. Yet, the ambiguity of the term leads many to wonder what is truly being referred to and what advances are truly available. Much of the rumors and confusion surrounding Cloud Computing could be solved by more clearly defining the term and what it encompasses. The first part of this thesis will describe Cloud Computing and its many forms. Then the thesis will describe the problem for which Cloud Computing could provide a solution. A potential model is described, created, and tested toward this effort.

Open source software, such as Ubuntu’s Linux operating system, provides an inexpensive alternative to the standard Windows and Mac operating systems found in many organizations. Linux itself is much more streamlined than either of its costly counterparts, and has much lower system requirements, making it a more responsive alternative as well [1]. Linux is also considered much safer from malware and other security risks than Windows [14]. This implies that not only could a move to open source software be cost effective, it could also improve productivity.

### **A. CLOUD COMPUTING DEFINITIONS**

It seems as if every company, website, community or author has its own definition for Cloud Computing. As the term is ambiguous by design, the prominence of a higher authority must be sought to clearly define the term for all parties. The National Institute of Standards and Technology (NIST) is an established authority in this case. They are responsible for developing standards and guidelines, including minimum requirements, for providing adequate information security for all government agency operations and assets [2]. NIST defines Cloud Computing as

A model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. [2]

NIST also defines three deployment models (see Figure 1) and four service models (see Figure 3) that make up this definition of cloud computing.

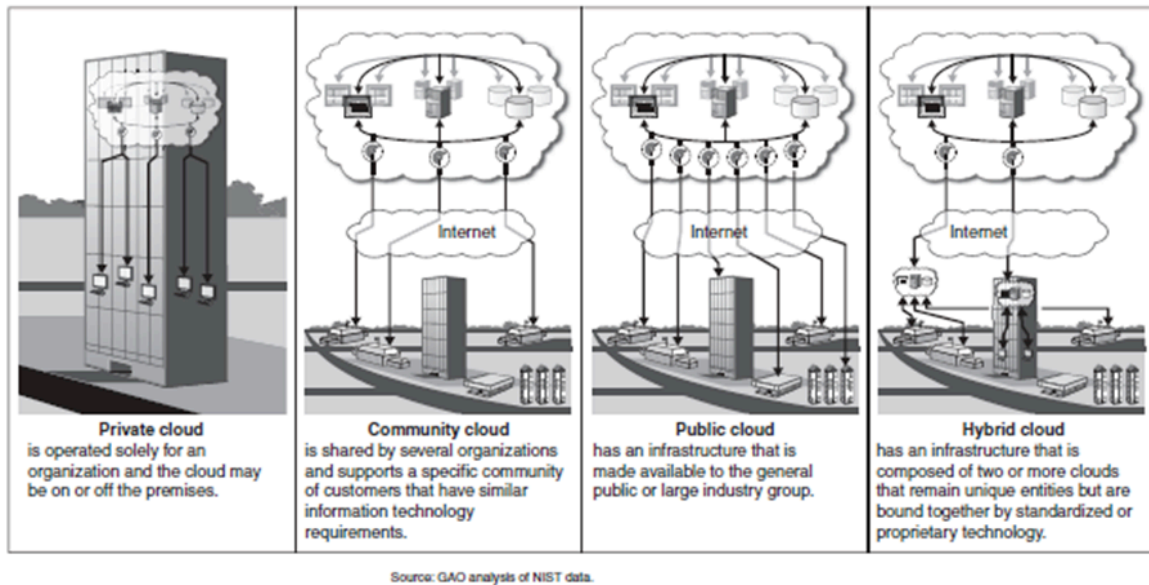


Figure 1. Cloud Computing Deployment Models [From 12]

## 1. Deployment Models [12]

Understanding the applications of Cloud Computing requires understanding the general models for implementation of Cloud Computing. The first of these is the “Deployment Models.” These models can be thought of as detailing the physical descriptions of the Cloud. This includes hardware specifications, software requirements, network design, user access methods, physical location and security. These include Public, Private and Hybrid clouds.

### *a. Public Cloud*

The public cloud is the most commonly used, and often mistaken as the only form of cloud computing. Access to a public cloud is possible anywhere there is an Internet connection, and is often available to the general public. The public cloud can be owned and operated by a business, college or government institution, and generally is physically located on the premises of the provider and not necessarily the customer, but

could be spread all over the globe. Public clouds can offer a wide variety of applications, from storage plans to shopping sites to virtualized desktops. The public cloud is a means of accessing institutional resources externally, for both employees and customers. The Amazon Marketplace is one example of a public cloud. Amazon is one of the leading cloud service providers. While they host their own business site, they also host cloud services for other organizations utilizing their Amazon Elastic Compute Cloud [11].

***b. Private Cloud***

In contrast to the Public Cloud, the Private Cloud is provisioned for exclusive use by the organization controlling it. A Private Cloud contains all the same aspects of the Public Cloud, but is not accessible by anyone outside the organization. In most cases, this type of Cloud is not even connected to the larger Internet. The Private Cloud is developed to allow a shared pool of resources to be utilized by a single organization or community. This means an organization can pool its resources together to provide a standardized environment for all of its employees, based on access level and job description. Private Clouds can be owned, managed and operated by the organization or by another party, or some combination of the two.

***c. Community Cloud***

A Community Cloud is similar to a Private Cloud, but is utilized by a community of consumers with shared concerns. It is basically a large Private Cloud which may be owned, managed and operated by one or more of the organizations in the community. The Community Cloud is generally accessed via an Internet connection like a Public Cloud, but not necessarily available to everyone.

***d. Hybrid Cloud***

A Hybrid Cloud is simply a combination of the other Cloud types. The Cloud infrastructure is a combination of two or more other cloud infrastructures, which remain separate entities connected to each other by some standardized technology which allows them to communicate and share information. This type of Cloud could be utilized

to allow public access to information stored in a Private or Community cloud. Most companies that employ cloud architectures involve some sort of Hybrid cloud (see Figure 2).

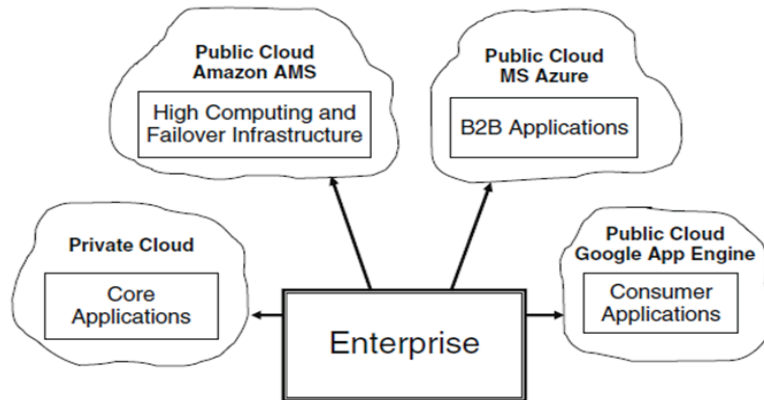


Figure 2. Illustration of Enterprise Cloud [From 1]

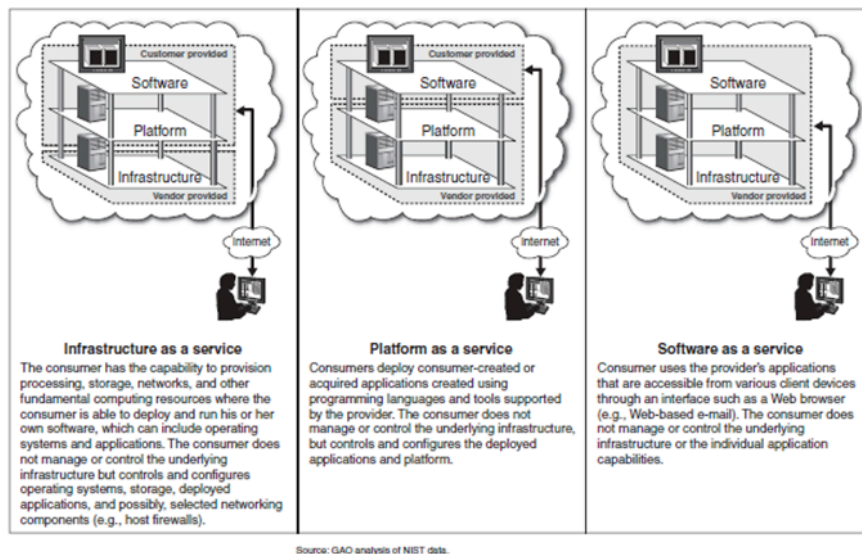


Figure 3. Cloud Computing Service Models [From 2]



## **2. Service Models [2]**

These models detail the usage of computing resources from the perspective of the Provider and consumer organization, not the individual end user. In a Public Cloud, the consumer would be the organization paying for the services hosted by the Cloud Provider.

### ***a. Software as a Service (SAAS)***

Software as a Service (SAAS) provides software applications to the consumer. Similar to Google Docs, the applications are accessible to the consumer through a web browser or other application interface. The consumer has no control over the underlying infrastructure, only access to the application. The consumer would then provide access to the applications they purchased to the end users, who may be employees or their own customers.

### ***b. Platform as a Service (PAAS)***

Platform as a Service (PAAS) provides the consumer with access to a platform on which the consumer may install applications. This means the applications available are chosen by the consumer, but the consumer still does not have access to the cloud infrastructure itself. The consumer will have control over the applications deployed and any application specific settings. The consumer would then provide access to these applications to the end users. The consumer does not have control of the hardware.

### ***c. Infrastructure as a Service (IAAS)***

Infrastructure as a Service (IAAS) provides to the consumer the ability to provision physical computing resources, such as processing, storage, networks and others, on which the consumer is then able to run arbitrary software, which in this case could include operating systems. Much more control is available in this model, as the consumer is able to control the operating environment, storage allocation and applications, and possibly network security. An IaaS model as a private cloud is the example used for the remainder of this thesis.

## **B. VIRTUALIZATION**

### **1. General Description**

The cornerstone that makes Cloud computing possible is the advancement in virtualization technology. Server hardware, and to a lesser extent software, has advanced to the point that servers are able to efficiently support multiple individual operating system instances, known as Virtual Machines (VM). Virtualization is the depiction of a physical system running as a self-contained software program. The program executes just as a physical machine would, and is capable of operating on specified physical hardware, but runs on top of the virtualization software itself.

The *Handbook of Cloud Computing* lists Virtualization as the primary enabling technology of cloud computing [4]. Virtualization is accomplished by providing a *Hypervisor* layer between the physical server and the guest operating systems, which allows multiple operating systems to be run as separate machines (see Figure 4). The hypervisor acts as a buffer between the physical hardware and the operating systems, while allowing the VMs access to the hardware itself with little or no interference from the server software. This reduces the number of physical machines required, and consolidates management of them. It also allows the number of machines, and the resources available, to be controlled automatically as demand changes.

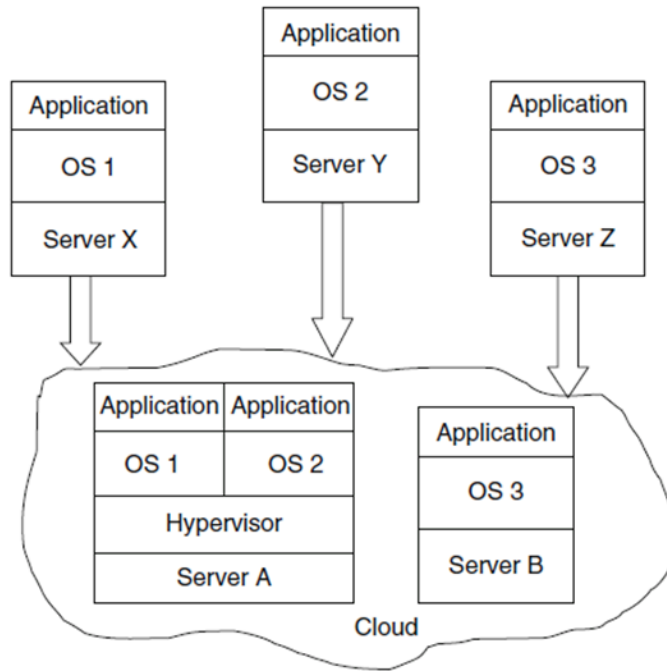


Figure 4. Illustration of Hypervisor Layer Virtualization [From 4]

## C. SUBMARINE NETWORKING

### 1. Types of Submarines

In understanding the types of submarine networks in service, it is first necessary to describe the types of submarines in service. There are two basic submarine platforms, the Fast Attack submarines consisting of the Los Angeles Class, Seawolf Class and Virginia Class; and Ohio Class submarines broken up into two types, SSBN ballistic missile submarines and SSGN guided missile submarines. These submarines all have significant design differences, which are important in understanding the network configurations.

#### a. *Fast Attack*

(1) Los Angeles Class. The Los Angeles-class attack submarine makes up the majority of active duty submarines, with 42 currently in service. These submarines

are aging and will eventually be replaced by the Virginia Class, but still have a long service life ahead of them. The last ones built in 1996 may see service into 2030 and beyond.

(2). Seawolf. While the Seawolf class may be the ugly stepsister of the submarine community, its extensive lifespan and capabilities will keep it an active participant for decades. As a significant change from the previous Los Angeles class, the Seawolf class saw many technological improvements, especially in terms of Information Technology.

(3). Virginia. The Virginia class was designed to replace the Los Angeles class and provide a cheaper alternative to the Seawolf class. While this may have come at the expense of some operating parameters, the advanced technology incorporated into the Virginia class was not one of them. The network on board the Virginia class is one of the most advanced in the Navy. The incorporation of commercial-off-the-shelf (COTS) products aided in this cost reduction, and adds to the ability of the network. The Virginia Class is slated to eventually receive the CANES network but not in the first iterations. It will be the only currently operational class of submarine to receive CANES, with the SSBNX also slated to whenever it is produced [10].

#### ***b. Ohio Class***

(1) SSBN. Products of the cold war, the Ohio Class SSBN submarines can carry 24 Nuclear missiles and are a vital part of the nuclear deterrence strategy. Constructed between 1976 and 1996, these submarines have some of the oldest and slowest data networks in service. As network communications are not a priority for ballistic submarines, refitting and upgrading the existing networks are also not a priority.

(2) SSGN. Four Ohio Class SSBNs were refitted to carry Tomahawk missiles instead of the nuclear missiles. These submarines were integrated as vital strike assets and completely overhauled in order to perform as Fast Attack submarines, while maintaining the benefits of size and crew turnover of the SSBN counterparts. During this process, the network systems were upgraded as well.

## **2. Basic Network Specifications**

### ***a. Hardware***

Submarine Network configurations are as numerous as the submarines they are installed aboard. The networks are often configured using a combination of COTS and military components from whatever source is available at the time of need. This is a problem common in the Navy, and one reason for the complexity of Navy networks. For simplicity sake, there are some common components to all submarine networks that we use as the basis for the network design presented within this research. These include: two Major server cabinets, two minor server cabinets, two main gigabit routers, and three main switches, all serving 35 user workstations with a total of 50 available network ports. The workstations may also vary, but in general and for the purposes of this thesis, are laptops.

### ***b. Software***

(1). Microsoft Windows 2000 Server, Exchange Server. The software backbone for the submarine network is a version of Microsoft server software, generally Windows 2000. A Microsoft Exchange server to accommodate the vital asset of e-mail communication complements the main server. The servers themselves are very similar in construction, containing a total of 1 terabyte of data and 16 gigabytes of RAM.

(2). OHMS-NG. A safety program for aiding in the electronic management of equipment tagouts, OHMS-NG has its own server running proprietary software. This is a smaller standalone server that could possibly be virtualized, but would simply provide database storage for the OHMS-NG program.

(3). Windows XP/Vista. All user workstations are running Windows XP and accessing user profiles stored on the Windows 2000 server. There is very limited access to the physical storage on the workstation itself, and generally users share workstations and will commonly access multiple workstations throughout the day.

## **D. LITERATURE REVIEW**

### **1. Cloud Computing Handbook**

The *Cloud Computing Handbook* is a collaborative effort among academic and corporate experts in the field of Cloud Computing. Released in 2010, the book's purpose is to "summarize cloud computing technologies, systems and architectures; cloud computing services; and a variety of cloud computing applications" [4]. The *Handbook* provides a concise, consolidated overview of cloud computing.

### **2. CANES**

The Consolidated Afloat Networks and Enterprise Services was designed to replace 5 shipboard legacy network programs [10]. Unfortunately, through FY13, the program will only be fielded on four classes of ships.. A consolidated afloat network that only applies to four ship platforms is not really consolidated. The price of this consolidation is surely the reason for the limited scope of the program.

### **3. NGEN**

Next Generation Network (NGEN) is the Navy's shore-side version of CANES. It is slated to replace the current Navy and Marine Corps Internet (NMCI) and has only recently been released in the form of a Request for Proposals, the initial step in procuring a new technology [8].

### **4. Quality of Service Cornerstones**

In *Cloud Computing: Principles, Systems and Applications*, the authors describe the four cornerstones of quality of service [7]. These are categories by which computer systems can be measured to ensure the proper level of satisfaction the organization will have from the systems. In analyzing these different categories, it is important to look at both how the system operates in general and how it specifically applies to the organization.

***a. Efficiency***

Efficiency measures how well the data traffic and latency are optimized. The authors argue that Latency is the most important factor affecting customer satisfaction with a system, and therefore it should have strictly specified limits and maintain them. They argue that this stems from data traffic, both internally and externally, which is one of the main cost factors in any computing framework, and reducing it should be a major goal.

***b. Scalability***

Scalability is dependent on the potential consumer base, and in the case of a Public cloud these platforms need to be able to scale on a massive level. This also includes the ability to scale rapidly during peak operating times, and support provisioning methods for popular customers at their key times. Scalability can also apply to the apportioned resources available to each specific end user. The VM instance one user accesses may have different resources available than the VM accessed by another user.

***c. Robustness***

Robustness determines the level of availability, effective use of redundancy and the ability to fail as gracefully as possible. For Public Clouds, when users are paying for services they expect a certain guaranteed operational percentage. For Private Clouds, users require access to their resources for their work. In either case, the probability of failure should be understood and appropriate methods for reinstating the service should be developed and clearly expressed.

***d. Security***

In any organization, proprietary information must be securely maintained. Provisions must exist to ensure both data and applications are protected from fraudulent access. One common misconception of cloud computing is that the data may not be secure, because it may not be possible to know where your data is stored at any given time. While this is true for large-scale Public Clouds, on the order of Amazon and

Google, the data for a given organization is generally stored locally. For Private Clouds, all data is stored locally and can be physically controlled as well.

## **E. PROBLEM MOTIVATION**

### **1. Frustration**

As with many great problems, the motivation for this research derives from frustration—the frustration with using excessively outdated IT components. The modern Navy exists on e-mail, PowerPoint and many other computer-required programs. In the typical day of a submarine officer, six hours are spent on watch, five hours sleeping and the rest is spent on a computer; studying, writing reports, scheduling, e-mailing, creating PowerPoint presentations, issuing tagouts, researching problems, giving training, and a seemingly endless list of other activities. Computer problems, while inherent with any network to some degree, are more prevalent in older systems. The combination of a Windows 2000 server, windows roaming desktops and laptops running Windows XP means that not only are the laptops themselves outdated and slow, but the software running on them is several iterations behind in network performance and server management.

#### ***a. Manpower Costs***

Life on a submarine is not what anyone would call “ideal.” It is a steel tube in the ocean, containing countless systems and interworking components. It is a complex machine, a literal system-of-systems. It is crowded, busy, complex, stressful and magnificent. There is constant need to move and find some place new as different events are scheduled in different spaces throughout the submarine. This means the laptops, which are limited in number, must also be moved. When using Windows XP and Windows 2000, moving laptops requires shutting off and restarting the computer, because the roaming desktops require a constant connection to the server. This takes time.



## **2. Compatibility**

Newer software programs and hardware equipment will simply not work with older versions of Windows. Windows XP is notoriously slow and encumbered by a plethora of issues [6]. This means software upgrades have to be conducted in outdated manners. New programs, which may be very popular and increase productivity, may simply not work at all. This stifles innovation and efficiency.

## **3. Standardization**

As mentioned before, submarine networks are as unique as the submarines themselves. While somewhat similar in design, they are different in the actual structure. This means standardization across the fleet is impossible, complicating upgrade and maintenance plans for the shore based support activities. The CANES network was envisioned to consolidate afloat networks, which would by definition standardize them. Unfortunately, since CANES only applies to a small handful of platforms, and at this time there are no submarines on that list, standardization is still a problem.

THIS PAGE INTENTIONALLY LEFT BLANK

## **II. PROBLEM DESCRIPTION**

Network architectures aboard U.S. Navy ships are old, outdated, and frustrating. The Navy has acknowledged this for some time, and is moving forward with the CANES program to install next generation networks on newer ships and platforms over the next decade. Unfortunately, CANES falls short of providing an alternative to the ships and platforms not selected. Budget constraints play a large role in this, along with the problems inherent in designing a system for multiple platforms. This means that the majority of Navy ships will be left with legacy networks, plagued by the same poor performance issues and data sharing problems which contribute to poor productivity, and left to fend for themselves while watching their counterparts upgrade to the newest and best equipment.

Nowhere is this more prevalent than in the submarine force. CANES will eventually come to the Virginia Class submarine, and perhaps some version will be aboard the SSBNX decades from now, but the vast majority of the submarines currently in operation will be stuck with the legacy systems they have now. This makes submarine networks a useful example to use when considering cloud computing and the benefits of streamlined software design for quality not quantity of computer usage.

### **A. HISTORY**

When the idea for CANES was first proposed, it was thought that somehow the Navy could possibly pull off a complete consolidation of its IT infrastructure. The standardization problem stems from combatant commanders, type commanders and ship commanders using their own discretionary money to finance, plan and implement networks to meet the increasing need, or desire, for ever greater connectivity. As personal computer networks have grown common over the last ten years, more people have an understanding of the basic requirements of network systems and how to set them up. This could lead people to falsely assume that setting up a network on board a ship is just as simple.

CANES was supposed to actually consolidate many of those networks, replacing both past and future equipment. Unfortunately, budget constraints have limited this to a select few platforms in the initial offering, leaving the rest of the Navy stuck with legacy equipment that is neither compatible nor standardized with the rest of the fleet. In many cases, ships are not capable of even communicating with one another because of minor differences in IT structure. What if it was possible to improve the rest of the fleet's IT infrastructure, without a major cost? The ideas presented here are not meant to replace or serve as an alternative to CANES. They are meant to complement CANES, by being applicable to all of the platforms not slated to receive the CANES program.

## **1. Client Server**

The evolution of computer systems has seen a reversal in many years, away from high-powered personal workstation back to a more Client/Server based architecture. In Client/Server systems, the Client computer accesses resources stored on the server. The types of resources, and the completeness of that access depend on the type of network philosophy utilized. It can range from simply shared storage to complete reliance on the server for all computer functions [5]. This same idea applies to Cloud Computing, and indeed many see Cloud Computing as an evolution of the Client/Server architecture. Cloud services can range from shared storage to complete control as well.

When utilizing Windows' Roaming Desktops, like most of the Navy utilizes, a constant connection to the server is required [4]. If that connection is lost, it can drastically affect the work being performed. Saving to the local storage on the workstation is not usually an option, as that feature is generally disabled for security and accountability reasons. Workstations are shared by many different people, so saving locally means others could access that information. The connection to the server must be reestablished to continue accessing the information. Generally during these connections the information is requested by the Client, retrieved by the Server, sent to the Client and then processed by the Client. When the client makes a change, that change is saved in a temporary cache on board the workstation, and then pushed back to the Server when

saved. The server simply acts as an intermediary in the process, and the communication is constantly going back and forth. This means the network connection is vital to the entire process.

## **2. Server Hardware**

The evolution of Client/Server architecture to the point that Cloud Computing has proliferated has more to do with the marked advancement in Server technology in recent years, as opposed to a user-friendliness for that type of setup [4]. Servers are capable of supporting many more clients now, with much better resources and processing power available to each of those clients. This is partly due to the advancement in server software, as well, and the development of sophisticated Hypervisors that serve as a backbone for virtualization. Hypervisors are a sophisticated software program, much like an operating system or possibly built into an operating system, which allow other programs to run as if they were running on the physical hardware itself. The hypervisors allow expansion, upgrades, and standardization without affecting the physical equipment itself. It is kind of like using a charging dock, which has multiple charging ports, for a cellphone. You can either plug the cellphone directly into the wall socket to charge it, or use the charging dock which is plugged into the same socket to charge multiple cell phones, which can be swapped out as needed. The Hypervisor layer, controlled by the Virtual Machine Manager, utilizes the physical server hardware to a near-optimized extent for the applications available.

## **3. Network Hardware**

Another advancement that has helped bring about the return to Client/Server setups is the incredibly high speeds most Local Area Networks (LANs) are capable of. With gigabit networks becoming common, and 10 and 40 gigabit finding their way into large datacenters, the amount of data that can be transferred across a network is quickly limited by the users rather than the network backbone. Most web applications need only a small trickle of that bandwidth, and even the Client/Server communication process needs only a small percentage [5]. This implies that the network infrastructure is vastly under-utilized.

#### **4. Clients**

One way in which the Client/Server architecture is improved upon is the number of devices which could act as clients. Clients can be made up of Thin, Thick and ZERO clients, as the Secure Shell program that accesses the virtualized instance is available on a variety of platforms. This provides for improved functionality across the board, because the client could be tailored to the specific needs of the user rather than the requirements of the server.

ZERO Clients are small computer workstations that lack the localized processing power found in a laptop or desktop computer. A ZERO Client is generally made up of a Network Interface Card (NIC), a Graphics Processing Unit (GPU) and inputs for peripheral devices like keyboards, mice and others. Thin Clients rely completely on the Server hardware to run the operating systems and applications in use, due to their lack of central processing ability and storage. The operating system image is transmitted over the network, and displayed on whatever is connected to the ZERO Client's GPU. This is accomplished mainly using virtual machines hosted on the Server, which the user accesses and controls from the Thin Client. A VM workstation of any type accessed from any client device would operate the same way.

Thick Clients are standard computer workstations, such as a desktop or laptop, which access the server through an application installed on that workstation. Thick Clients could be any device with another operating system installed on it, such as Windows, Mac, iOS or Android. The standardization network protocols allows for the cross-platform integration. This means a visitor with network access could use their own device tailored to their needs to access the shared resources available at the place they are visiting.

Like all models, there are some benefits and some drawbacks to this type of computer system. Not having any localized processing capability makes the network connection much more important, essentially vital. If the network is down, there is no way to access the Server's operating systems, and no work can be completed.

With the Client/Server model, the Navy has limited itself to a select number of clients which can access the shared resources. Using the roaming desktop model, accessing a Windows 2000 server from a windows PC means that no other types of clients or computing devices could be used. This means utilizing Tablet computers, Personal Digital Assistants, smart phones and other network capable devices is precluded. This may mean fewer headaches for network security, but it suppresses innovation and productivity improvements, as well as purchasing options when planning the network.

In a Cloud Computing environment, access to the resources is the main function. This means that the more access methods there are, the more desirable the model is. Amazon's cloud can be accessed through any device with an Internet connection, as is the case for most clouds. Clouds are built with standard Internet Protocols in mind, so that as many people and devices as possible can utilize the resources.

## **B. CONTEXT**

### **1. Confined Area**

By definition, a submarine is a confined area, limited in both size and configuration. This confinement goes beyond the physical construction, though. When operating at sea, under water, a submarine is a completely enclosed eco-system, isolated often for days on end with little or no access to the outside world. It is confined to itself, relying on the resources it is equipped with. This means that more than any other platform the submarine must rely on itself for daily operations. In the IT department, it means the submarine is not just limited to slow Internet access during peak times, or random spots of bad connectivity. The submarine is limited to no connectivity to the outside world.

Connectivity seems vital to the Navy recently. Video conferencing with captains at sea is common, understanding the Common Operational Picture (COP) is pushed to the lowest level possible. These ideas are not found in the submarine community. The resources a submarine has when it goes to sea are what it will have the entire time, and they must be shared on board to the advantage of the submarine as a whole. When the

Captain of a submarine has the same access to computing resources as the newest, most junior crewmember, while it may be humbling for those involved, it limits the full range of command and control available to that Captain. A network that could dynamically allocate resources and processing power based on a set number of priority levels would be incredibly beneficial.

## **2. Standardized Equipment and Procedures**

According to the SPAWAR PEO C4I website, the Navy's Submarine Local Area Network (SUBLAN):

Provides Navy submarines with reliable high-speed secret, sensitive but unclassified and top secret Local Area Networks. When the SubLAN network is combined with other subsystems, it delivers an end-to-end netcentric warfare capability. AN/USQ-177 Variants (V)1,2,3,4 provide network infrastructure including an Unclassified Wireless Local Area Network (UWLAN), servers, and the Common PC Operating System Environment, which provides the server and operating system environment for other applications such as Non-Tactical Data Processing System. [13]

Using a submarine for this study simplifies the discussion of network hardware, because in general submarines have the same basic network structure, provided by the SUBLAN program. Submarines also have the same basic command structure, personnel assignments and manning, unlike the many configurations and crew compliments of the diverse surface fleet. This simplifies things even more, because different aspects of the command can be analyzed as to how they would be affected by the network.

## **3. Air-Gap**

Classification of information stored aboard a submarine is critical. In general, there is little or no unclassified network on board submarines, mainly because a submarine is an enclosed classified storage area, where little or no unclassified information is used. The separation between the two is accomplished using security protocols, but mainly through the physical air gap between the classified and unclassified



networks. Access to classified information is not possible through an unclassified network because they are not physically connected anywhere.

The air-gap provides a wonderful model for using cloud computing. There are two different air gaps to be considered. The first is the local security classification air gap. It allows for many of the security problems to be negated, and security is the most common complaint about cloud computing. The second air gap is between the submarine and the outside world, which also provides a great transition from public to private clouds possible. A submarine has no reason for a public cloud, because the submarine will not be able to connect to it while at sea.

#### **4. Budget Constraints, Older Platforms**

Budget constraints are a major problem plaguing all services during this time period. This means that if a program is not cut completely, its scope will be shrunk at the least. There may be fewer units purchased, or fewer modifications made, or fewer platforms planned. This is the case for the Navy's CANES program. It leaves the majority of the currently operating ship platforms without hope of upgraded network hardware, while at the same time expecting more technologically advanced interactions out of the fleet. Utilizing an open-source private cloud could improve productivity and submarine network capabilities at a fraction of the cost of a full CANES upgrade.

THIS PAGE INTENTIONALLY LEFT BLANK

### **III. MODEL DESCRIPTION**

#### **A. ASSUMPTIONS**

To best describe the model proposed as a possible network solution, it is necessary to look at it from the NIST definition mentioned in Chapter 1. The first part of that definition is “a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources” [2]. This means that the network model should provide access to the shared resources used by the entire organization, from anywhere a user might need that information and make that access simple and user-friendly.

The shared resources used on a submarine are similar to those found in many organizations. Typically training materials, such as PowerPoint presentations, videos and other software applications, take up the largest portion of this storage. Publications and doctrine files are another large portion of the shared materials held in shared storage.

Providing access to these resources would benefit everyone on board, and possibly save quite a bit of storage space since the resources would be stored together rather than held separately in different locations. To benefit from the sharing of these resources, all crew members would have to be able to access the materials wherever they needed that access, or at least wherever access was typically made, making it ubiquitous.

Making access to the materials user friendly is where the Ubuntu Private Cloud really shines. Ubuntu is an open-source project based on the Linux operating system kernel. The versions used in this research are from the 10.04 update, which uses the same Elastic Computing 2 (EC2) virtualization method that many large corporations, such as Amazon, use for their cloud computing needs [9]. The differences are a matter of scale and usage. Amazon uses a cloud architecture in order to serve hundreds of millions of customers a variety of resources. They provide web businesses, storage and other virtualized environments. While similar, the submarine does not need its systems to be as robust. Providing a standardized operating environment with access to shared resources is enough.

At the basic level, the Ubuntu Private Cloud would replace the Windows 2000 Server operating system, the Windows XP/Vista operating systems on the workstations and the Windows Roaming Desktops used to access the network. Using the Ubuntu software improves the network performance in two basic ways: the Ubuntu operating system has better performance characteristics than Windows XP or Vista partially due to its lower resource requirements, and the Ubuntu Private Cloud does not suffer from the network problems that the Windows Roaming Desktop experiences. The Ubuntu Private Cloud is a Platform as a Service (PAAS) model which provides the basic server hardware and software upon which the user experience is built. In this case, the cloud presents a virtualized operating system. The thesis presented here looks at a basic server model, and then extrapolates that out to two real world examples.

## **1. Hardware**

### ***a. Experimental Model***

The first model used in the research was a very small-scale model constructed with three older-model laptops and a typical home networking router (see Figure 5). These laptops came originally installed with either Windows XP or Windows Vista, but have since been wiped and only the Ubuntu software installed. The key component in this setup was the presence of Virtualization technology on the hardware itself. In order to take advantage of advanced Cloud Computing features, the hardware used, in this case the motherboard and processor, are required to have virtualization built in. As recent as 2009, virtualization technology was really only present in higher-end personal computers, but is becoming much more common now as Cloud Computing is becoming more prevalent.

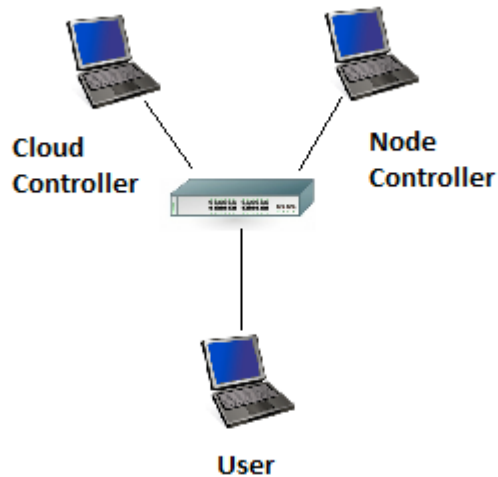


Figure 5. Experimental Cloud Setup

***b. Submarine***

The next model is the first one to be extrapolated from the original. Using the standardized equipment mentioned in Chapter II, this is what the standard submarine network could look like using the Ubuntu Private Cloud server software (see Figure 6). The model is limited by number of workstations available on board, not by the server hardware itself. Since the software is scalable, as was seen in the first example, the sub model could be scaled to whatever specific hardware is present on board a specific submarine, to accurately gauge the capabilities.

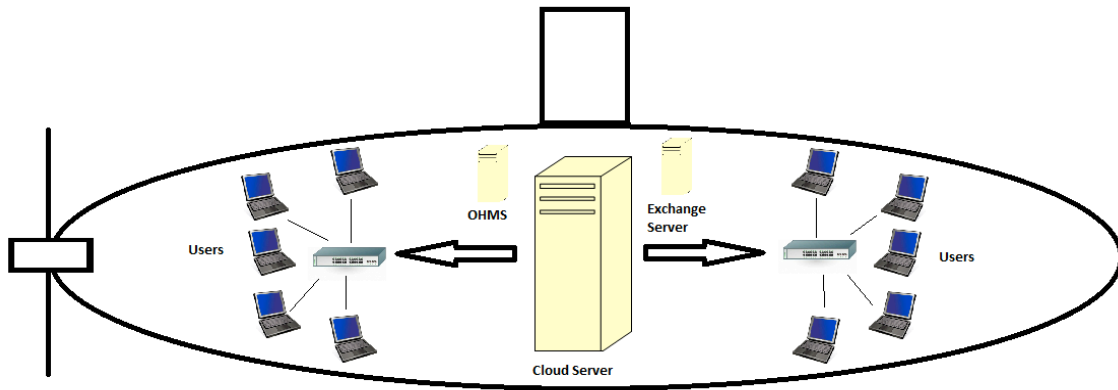


Figure 6. Submarine Cloud Model

### *c. Squadron/Group*

Submarines are grouped together in Squadrons based on mission types and location. A Commodore, who is a former submarine commander, commands a Squadron. The squadrons provide logistical, training and coordination services for the submarines. Squadrons are collected together into Groups, which are commanded by a Rear Admiral. Groups are also generally based on geographic location, but there are examples of groups for mission types such as Group 9 until recently commanded all Ohio-class submarines.

The network infrastructure for a Squadron or Group would be much different than on board a submarine, but there is no reason for the model to be different in anything other than scale. The virtualization software could easily scale to provide access for thousands of personnel over large areas, based solely on the network infrastructure and server hardware present.

## **2. Software**

The software used in all of these models is based around Ubuntu's version 10.04 of both their server and desktop operating systems. The desktop operating system is installed on the workstations, and is used as the virtualized environment accessed on the server. The server software is very minimalistic. It does not have a flashy user interface

like Microsoft's Windows Aero interface. Control is affected through a command prompt where the user types strings of commands to perform various functions.

The Ubuntu cloud provides VM instances of various sizes and in quantities directly dependent on the hardware specifications utilized. The number of running instances can be controlled by the network administrators to ensure resources are utilized by those with higher priority, such as the Captain or Officer of the Deck or whoever is leading a training session. The following table shows the size of the available VMs as well as the simulated hardware specifications these VMs would have. The VMs do actually utilize the hardware directly, through the hypervisor layer, making them run as if they were physical machines with these specifications. These are the default sizes but can be modified by the administrator [9].

<b>Size</b>	<b>CPU (Processors)</b>	<b>RAM (MB)</b>	<b>DISK(storage, GB)</b>
Small	1	192	2
Medium	1	256	5
Large	2	512	10
X-Large	2	1024	20
XX-Large	4	2048	20

Table 1. Virtual Machine Instance Sizes

### **3. Budget**

For the purposes of this thesis, the budget for this upgrade process is assumed to be minimal, if not zero. Ubuntu's operating systems are open source, and therefore free at the basic level. Where Ubuntu makes their money is in providing technical support and service to corporations using their software. This is generally on a subscription base, with plans varying in size depending on the amount and type of service desired. We will cover this briefly in Chapter 5, but chose to negate budget in this thesis because we have confidence in the technical abilities of submarine crew members and note that as we

have taught ourselves how to operate the system, a training plan could be developed to reach self-sustaining technical proficiency of the systems without requiring any assistance from the Ubuntu technicians.

## **B. OUTPUTS AND FINDINGS**

Measuring the improvement to the user experience would be difficult without real world conditions that are impossible to simulate outside of an actual submarine. The confinements of a small laboratory, while similar to the confinements of a submarine, are not quite as harsh of an environment. We chose a few factors that measure general user satisfaction levels in order to gauge the effect the software changes would have on common daily tasks. Workstation startup time, network connection time, VM load time, time to recovery from loss of connection, server reboot time were chosen to gauge a general comparison of computer performance factors which have a common effect on user satisfaction based on the usability of the computer systems. These were measured by timing the events before and after installing the Ubuntu software.

<b>Steps</b>	<b>Original</b>	<b>Ubuntu</b>
Workstation Startup Time	3.2Mins	.59Mins
Network Connection Time	.55Mins	.23Mins
VM Load Time	N/A	0.78Mins
Time to recovery from loss of connection	2.1Mins	0.84Mins
Server Reboot time	N/A	0.77Mins

Table 2. Operational Time Requirements (in Minutes)

## **C. OPERATIONS**

### **1. Lab Setup Process**

The Ubuntu server setup process is fairly straight forward. The following table lists the basic steps taken for the server install and the time required for each step. The



information can be found in several resources online as well as on Ubuntu's website itself, but for this setup process I used a well-illustrated document available online [9].

Steps	Exp.
1. Install Ubuntu Enterprise Cloud: Cloud Controller	2.19hrs
2. Install Node Controller	1.44hrs
3. Install euca2ools from admin console	0.23hrs
4. Install an Operating System Image	0.11hrs
5. Run an instance of the Operating System	0.03hrs

Table 3. Ubuntu Setup Process with Required Times

Installing the Ubuntu Cloud Controller is similar to installing any other operating system. You start by inserting the disk and run through menu choices for region, language and keyboard layout. The program then goes through detecting available hardware, configuring the network, naming the Cloud Controller, partitioning the hard drives and finally selecting the options for installation. For the Cloud Controller with Ubuntu, it is possible to install the Cloud Controller server, Walrus storage service, Cluster controller and Storage controller all on one physical server. The Node Controller must be a separate physical machine.

Installing the Node Controller is almost identical to installing the Cloud Controller, with the exception of the options picked prior to installation. For the Node Controller, only the option for Node Controller is selected. The Node Controller is the machine which will actually run the VMs, so it is important for it to have the best specifications.

Euca2ools is a program run on the admin console in order to manage the various VMs installed in the cloud. It is accessed from any web browser with access to the network the cloud is on, through an admin site. This would allow maintenance to be performed by a network administrator from anywhere with access to the network.

Once Euca2ools is installed, selecting an operating system to install is accomplished via the web interface by selecting the links for the desired version. The software comes with Ubuntu's own Desktop operating systems, but more advanced users could package their own operating system platforms [9].

An instance is then run from the command line of the Cloud Controller server. Once the instance is running, users may log into it from any network connection with access to the cloud.

## 2. Accessibility

Accessibility is measured by the number of instances available and where those instances may be accessed from. The numbers listed in the output tables are based on the number of simultaneously accessible machines.

## D. DATA

### 1. Experimental model

#### a. *Hardware Specifications*

Hardware	Cloud Controller	Node Controller	Workstation
Manufacturer	HP	Asus	Dell
CPU	3GHz Dual Core	1.7Ghz Quad Core	1.9Ghz Centrino
RAM	4GB	8GB	2GB
Storage	250GB	1000GB	320GB

Table 4. Experimental Model Hardware Specifications

***b. Outputs***

Size	Number
Small	8
Medium	8
Large	4
X-Large	4
XX-Large	2

Table 5. Experimental Model Output

***c. Accessibility***

In the Experimental model, the cloud could be accessed both wirelessly and directly within a Wi-Fi connection. The Ubuntu software makes an old Dell laptop which we used as a workstation act like a brand new computer, partly because of the lower operating requirements.

**2. Potential Sub Model**

***a. Hardware Specifications***

Hardware	Cloud Controller	Node Controller	Workstation
Manufacturer	Dell	Dell	Dell
CPU	Xeon Quad Core	Xeon Quad Core	1.7GHz Pentium
RAM	16GB	16GB	2GB
Storage	1 Terabyte	1 Terabyte	200GB

Table 6. Potential Sub Model Specifications

***b. Outputs***

Size	Number
Small	32
Medium	32
Large	16
X-Large	16
XX-Large	8

Table 7. Potential Submarine Model Output

***c. Accessibility***

The VMs would be accessible from anywhere on the submarine's network. The number of small/medium VMs is equivalent to the number of workstations on the typical submarine, if all were used simultaneously, which could happen throughout a busy work day. These VMs would provide the same familiar user interface installed on the workstations themselves, while allowing access to remotely stored shared resources.

**3. Potential Squadron/Group Model**

***a. Hardware Specifications***

Hardware	Cloud Controller	Node Controller	Workstation
Manufacturer	Intel	Intel	Dell
CPU	Xeon Quad Core	Xeon Quad Core	2Ghz Dual Core
RAM	32GB	32GB	2GB
Storage	5 Terabyte	5 Terabyte	500GB

Table 8. Potential Squadron/Group Model

***b. Outputs***

Size	Number
Small	64
Medium	64
Large	32
X-Large	32
XX-Large	16

Table 9. Potential Squadron/Group Output

***c. Accessibility***

Accessing the Squadron or Group network from anywhere on the local network would allow access to the same VM interface at any workstation. The sub and Squadron clouds would be using the same basic server setups and would easily form a Hybrid cloud when the submarine is in port, so that access to the larger resources on shore could be accomplished from the submarine network. All of these models are based on one physical server blade used for both the Cloud Controller and the Node Controller. Adding more Node Controllers is simple and can increase the number of available VMs.

THIS PAGE INTENTIONALLY LEFT BLANK

## **IV. CONCLUSION**

### **A. PRIVATE CLOUD PARADIGM**

#### **1. What Cloud Means**

Understanding what a Cloud Computing platform can provide, beyond simple remote storage and online shopping, is essential to understanding the true potential. Cloud Computing is a shift from the typical Client/Server architecture, where the client is dependent on the server for most of the processes it runs, to a Server-centric architecture, where all of the resources are hosted and processed on the server. The user can access that information from whatever supported device is available and see the exact same data in the same format.

Cloud platforms provide the most efficient method for maximizing efficiency of shared resources. It utilizes the virtualization capabilities of a server to provide standardized, shared resources to the end user. Those resources are easily managed from a single central location, but can be accessed from any network access point. Moving past the misunderstanding that Cloud Computing is only really applicable on a large scale allows greater flexibility in network planning and infrastructure management. Just as physical clouds come in all shapes and sizes, Cloud Computing infrastructures can vary from a small home network with a handful of users, to global corporations with millions of customers. The limits are based on need and hardware availability, not size. A Cloud Computing environment on board a submarine could provide a good example of how a limited Private Cloud architecture could take the resources available and utilize them more efficiently.

### **B. POTENTIAL SOLUTIONS**

#### **1. Purchase New Equipment**

Purchasing new equipment is costly and not practical in an era of fiscal savings. The majority of the ideas presented here could be accomplished with little or no cost. New hardware could be purchased for all submarines to improve the existing performance while maintaining the current operating environment. The new hardware

would perform better, but some cost could be saved if only the servers were upgraded. Upgrading the server hardware, and then using a different VM interface such as Windows' Hyper-V server, could provide the same performance results without leaving the comfort and familiarity of the Windows operating system.

## **2. Stick with Current Equipment and Operations**

Staying with the current equipment and functionality does not seem like an option to most commanders, who seem to want more and more access to the outside world and the computing power out there. While many information system requirements stem from the need for video conferencing and live image analysis, that type of data transfer problem is not solved by switching to a cloud interface. It is solved by upgrading the external communication methods to allow more bandwidth utilization. The cloud platform offers better utilization of the equipment available, in order to improve productivity.

## **3. Improvise**

Improvising with the setups currently in operation to make a standardized system with improved performance may be the best option. This is where the cloud platform comes in. Utilizing a cloud architecture could improve productivity without requiring a great expense.

# **C. QUALITY OF SERVICE CORNERSTONES**

## **1. Efficiency**

Analyzing the submarine model on efficiency would require quite a bit of extensive testing equipment to measure latency and bandwidth management on real world equipment. Simply stating that it is more efficient is not sufficient to understand how the efficiency is measured. In this case, the efficiency is judged on the difference between the communications taking place on the network during a Roaming Desktop session and a FTP session accessing a server-based VM. During the Roaming Desktop session, the workstation is constantly accessing the data stored on the server, the server transfers that data to the workstation, the workstation performs the processing and



manipulation, then resends that data back across the network. Since all processing, data manipulation and application functions are performed on the server during the FTP session, and only viewed from the workstation, the communications required over the network are a much simpler cause and effect event.

## **2. Scalability**

The difference between the first two models show how scalable the Private Cloud could be. With no changes in software costs or upgrades, the Ubuntu server operating system provides the most possible virtual machine environments based on the physical resources available. This means the same software package, with the same knowledge requirements and maintenance needs, could be installed on any network size from the individual submarine to the entire Department of Defense.

## **3. Robustness**

Cloud Computing provides for a constant state of redundancy. Mirrored servers provide ready backups. Mirrored storage provides a backup of data in near real-time. The ability to scale the number of Node Controllers means the same level of access is maintained and user experience remains the same. In the event of a network interruption, which would cause a Roaming Desktop on Windows 2000 to require a restart, the SSL connection is simply reestablished to the VM instance that has been running uninterrupted on the server. This could reduce the risk of data loss and frustration from a delayed recovery.

## **4. Security**

Private Clouds benefit from the same security measures that any air-gapped network would benefit from. While access to an outside network is possible, and clouds can be combined to form hybrid clouds, access is still based on the same user levels. The concerns voiced over data storage security are not valid since the data is stored locally on centralized servers. Private Clouds have been touted as a secure method for implementing cloud computing.

#### **D. SUMMARY**

In summary, cloud computing could provide a complimentary alternative to the Consolidated Afloat Network and Enterprise Systems (CANES) program and improve both productivity and functionality of the fleet's Information Technology infrastructure. The computer networks on board submarines provide a good example of how cloud computing could benefit older platforms. This thesis presented a proof of concept for the use of a Private Cloud architecture on board U.S. submarines and how improving computer networks may be possible without completely reconfiguring or replacing the hardware, instead leveraging currently installed hardware. We used the open source "Ubuntu Server Private Cloud" as a basic example to illustrate and explore potential benefits and limitations of the Platform as a Service (PAAS) model. The revised system was examined in terms of its application aboard a submarine and how it compares to previous network architectures, such as the Client/Server model. Private clouds could improve productivity without affecting budget concerns.

## **V. RECOMMENDATIONS AND FUTURE RESEARCH**

The scope of this thesis has focused on the proof of concept of a Private Cloud architecture on board a U.S. submarine as an example of how Private Cloud Computing could be put to use to improve legacy hardware. We focused on defining what Cloud Computing was, in order to define what a Private Cloud was and how it could be used in this instance. The study of Cloud Computing is receiving wide ranging academic, media and commercial attention, because it has become the new sexy computing model of the near future. Cloud Computing encompasses a wide range of computing strategies, and covering them all in one thesis would be impossible. There are two general areas that this type of research could be further explored, which were far out of my realm of expertise; an analysis of the cost of this upgrade and a look at classified systems and measurements.

### **A. COST ANALYSIS**

#### **1. Virtualization**

The example in this thesis involves using Ubuntu version 10.04 Server and Workstation operating systems. Both of these operating systems are available as free downloads at Ubuntu's website. The Ubuntu workstation version was used as the VM model, because it was free, but the Cloud has the ability to host Windows operating systems and applications as well. An analysis could be conducted on the costs and processes needed to make a Windows VM run, and what it would cost to host Windows VMs as the primary user access method.

#### **2. Technical support Subscriptions**

Ubuntu offers various levels of technical support. An analysis could be done on the types of support offered, as well as coordinating with Ubuntu for specific cost plans and the feasibility of accomplishing that technical support. Also, training courses are provided on Ubuntu's website, but training a large number of IT personnel throughout the fleet may prove more costly. An analysis could be performed on the cost of manpower and logistics to train current IT personnel in Linux based software.

## **B. CLASSIFICATION**

### **1. Specific Equipment**

The specific equipment on board the submarines is important because it would give a basic idea of the VMs the servers would be able to host. It would also determine whether or not the servers had the actual Virtualization technology built in. More realistic testing could be done with actual equipment, but this could also require more funding for the experiments and the lab itself.

### **2. Security Analysis**

A specific analysis done of the security vulnerabilities in a Private Cloud could take an extensive amount of work and probably span several thesis projects. While the biggest security barrier for all classified networks is the air gap between classified and unclassified material, there are still instances where the data is encrypted and sent over commercial networks. This is not the case on board the submarine, but a similar examination of the ability to secure the VM access could be conducted. An analysis of the Secure Socket Layer used to access the VMs could also be conducted to determine its strengths and weaknesses.

## LIST OF REFERENCES

- [1] Subcommittee on Government Management, “Cloud computing: Benefits and risks of moving federal IT to the cloud,” in Joint Hearing 111–79, Washington, 2010.
- [2] P. Mell and T. Grance, “The NIST definition of cloud computing,” National Institute for Standards and Technology, Gaithersburg, MD, 2011.
- [3] V. Kundra, *Federal Cloud Computing Strategy*, White House Office of the Chief Information Officer, Washington, 2011.
- [4] B. Furht and A. Escalante, *Handbook of Cloud Computing*. New York: Springer Science+Business Media, 2010.
- [5] W. Y. Chang, H. Abu-Amara and J. F. Sanford, *Transforming Enterprise Cloud Services*. Irvine: Springer Science+Business Media, 2010.
- [6] R. Broida, “Linux vs. Windows: Should your office make the switch?,” CBS Money Watch, 22 June 2011. [Online]. Available: [http://www.cbsnews.com/8301-505143\\_162-28651688/linux-vs-windows-should-your-office-make-the-switch/](http://www.cbsnews.com/8301-505143_162-28651688/linux-vs-windows-should-your-office-make-the-switch/) [accessed 12 February 2012].
- [7] N. Antonopoulos and L. Gillam, *Cloud Computing: Principles, Systems and Applications*, London: Springer-Verlag London Limited, 2010.
- [8] “NGEN RFP Released,” Department of the Navy Chief Information Officer, 9 May 2012 [Online]. Available: <http://www.doncio.navy.mil/ContentView.aspx?ID=3961> [accessed 29 May 2012].
- [9] D. Animesh, “Build your own private cloud using Ubuntu 10.04,” 2011 [Online]. Available: <http://animeshdas.files.wordpress.com/2012/02/buildyourownprivatecloudusingubuntu10-04eucalyptusenterprisecloudcomputingplatform-by-animesh-das.pdf>. [accessed 12 April 2012].
- [10] “U.S. Navy leaning on CANES to Integrate shipboard networks,” Defense Industry Daily, 08 March 2012 [Online]. Available: <http://www.defenseindustrydaily.com/U.S.-navy-to-lean-on-canes-to-integrate-shipboard-networks-06221/> [accessed 15 April 2012].
- [11] Amazon EC2 Cloud. Available: <http://aws.amazon.com/ec2/> [accessed 15 April 2012].

- [12] Government Accountability Office, “Information Security: Federal Guidance needed to address control issues with implementing cloud computing,” U.S. Government Accountability Office, Washington, 2012.
- [13] PEO C4I Products and Services. Program Executive Office for Command, Control, Communications, Computers and Intelligence. Available: <http://www.public.navy.mil/spawar/PEOC4I/ProductsServices/Pages/default.aspx> [accessed 15 April 2012].
- [14] Parrish, Kevin. “Dell: Ubuntu safer than Windows.” Tom’s Hardware, 14 June 2010 [Online]. Available: <http://www.public.navy.mil/spawar/PEOC4I/ProductsServices/Pages/default.aspx> [accessed 15 April 2012].

## INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center  
Ft. Belvoir, VA
2. Dudley Knox Library  
Naval Postgraduate School  
Monterey, CA
3. Douglas MacKinnon  
Naval Postgraduate School  
Monterey, CA
4. Albert Barreto  
Naval Postgraduate School  
Monterey, CA
5. Christopher Brooks  
Naval Postgraduate School  
Monterey, CA
6. Dan Boger  
Naval Postgraduate School  
Monterey, CA